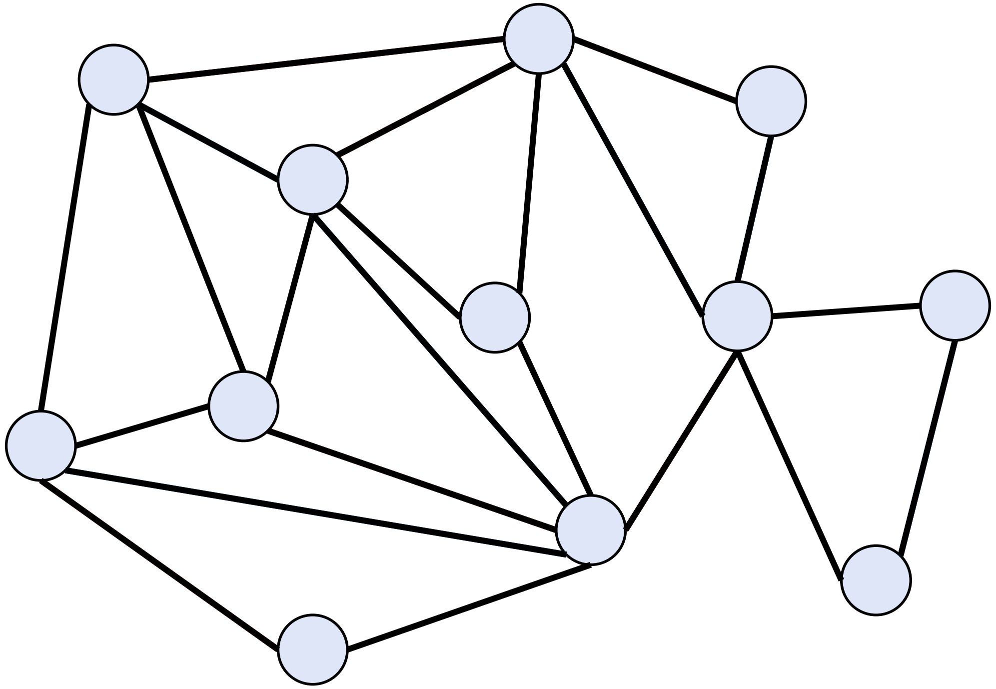
CPSC 768: Scalable and Private Graph Algorithms

Lecture 5: Approximate Average Degree in the Sublinear Model; Arboricity and Orientation

Quanquan C. Liu
quanquan.liu@yale.edu

Outdegree Orientation

- $(1 + \varepsilon)$-approx. for average degree + useful graph property!



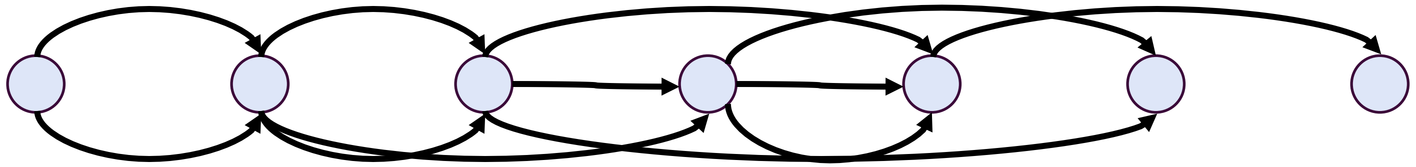
Orient all edges from low to high degree, what's the max out-degree that you see?

Give an Ordering of the Vertices to Minimize Outdegree

- Order the vertices of the graph in total ordering

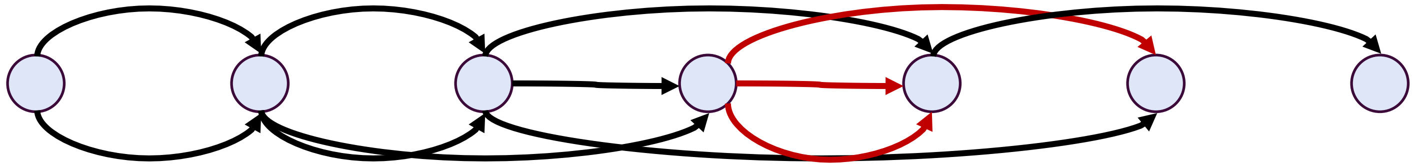
Give an Ordering of the Vertices to Minimize Outdegree

- Order the vertices of the graph in total ordering
- Orient the edges from vertices earlier in the ordering to later



Give an Ordering of the Vertices to Minimize Outdegree

- Order the vertices of the graph in total ordering
- Orient the edges from vertices earlier in the ordering to later

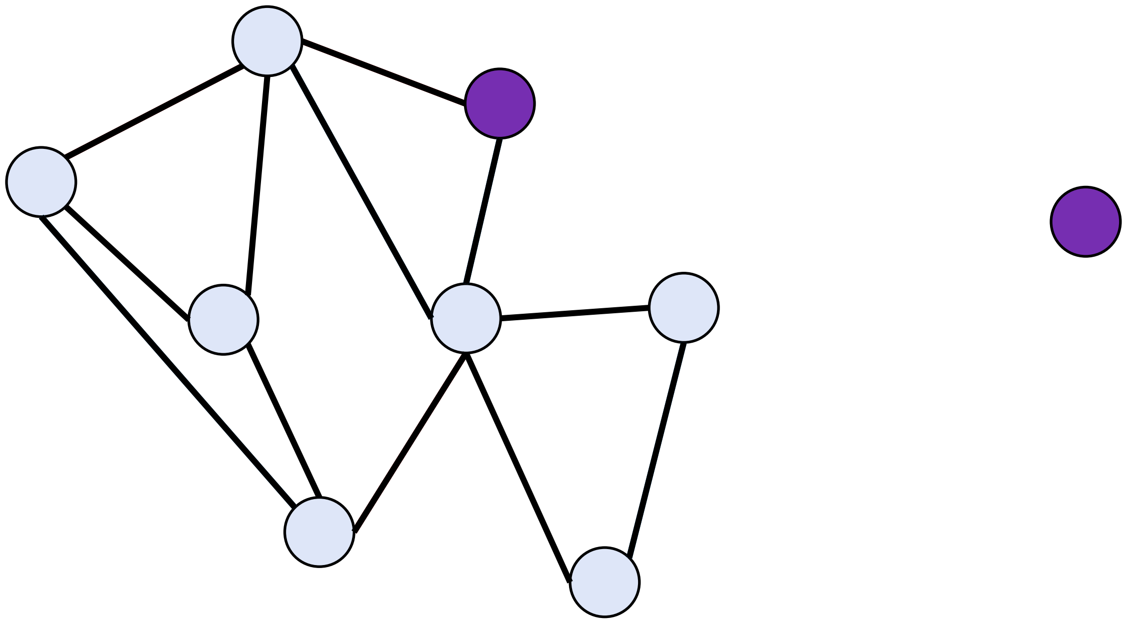


- Produces an ordering that minimizes the *maximum* outdegree

Ideas for an algorithm to achieve this?

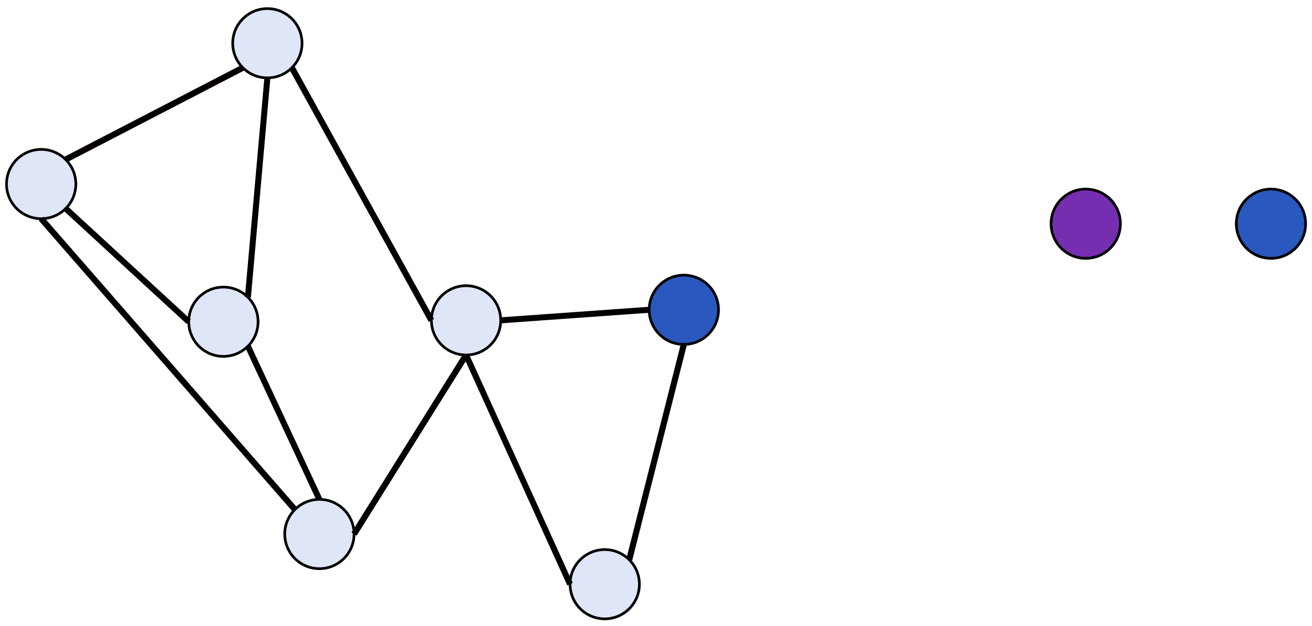
Linear Time $O(n + m)$ Solution

- Repeated peel vertex with minimum remaining induced degree and put in order



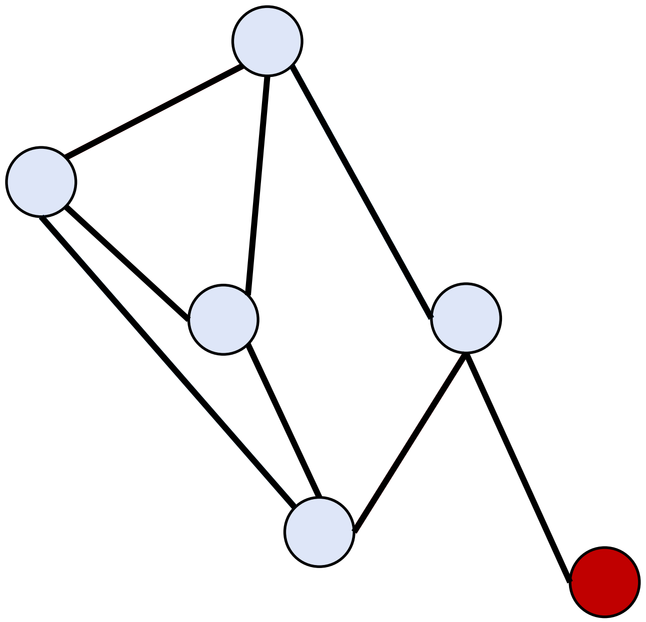
Linear Time $O(n + m)$ Solution

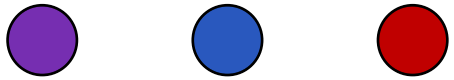
- Repeated peel vertex with minimum remaining induced degree and put in order



Linear Time $O(n + m)$ Solution

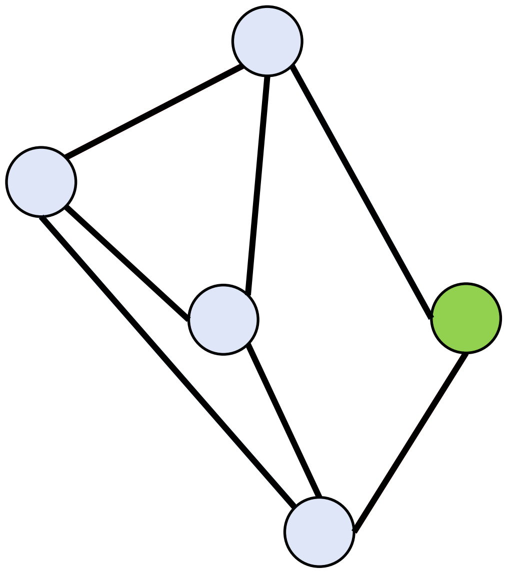
- Repeated peel vertex with minimum remaining induced degree and put in order





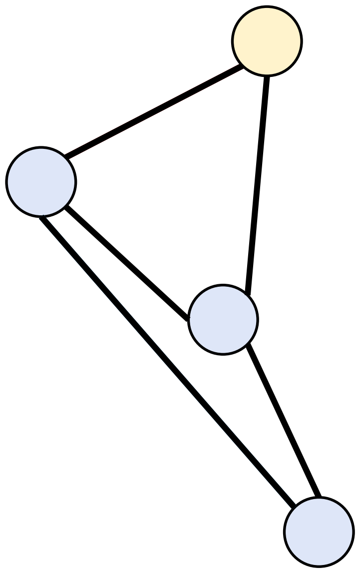
Linear Time $O(n + m)$ Solution

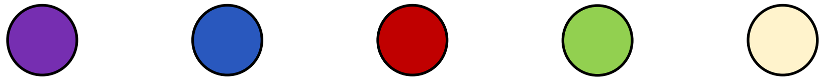
- Repeated peel vertex with minimum remaining induced degree and put in order



Linear Time $O(n + m)$ Solution

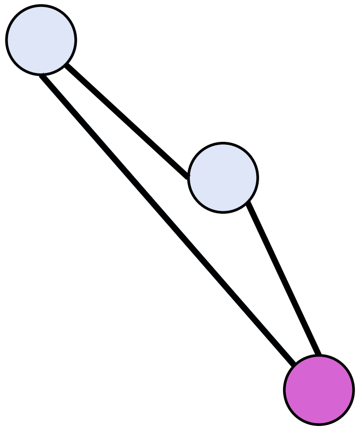
- Repeated peel vertex with minimum remaining induced degree and put in order





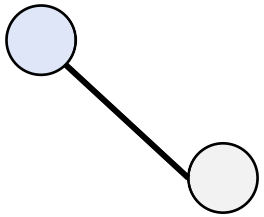
Linear Time $O(n + m)$ Solution

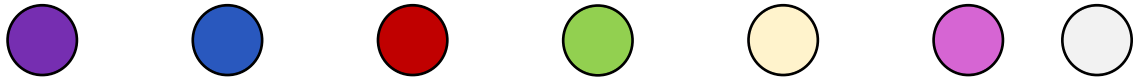
- Repeated peel vertex with minimum remaining induced degree and put in order



Linear Time $O(n + m)$ Solution

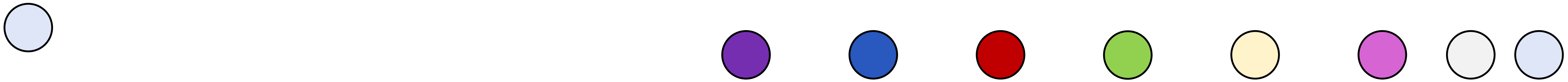
- Repeated peel vertex with minimum remaining induced degree and put in order





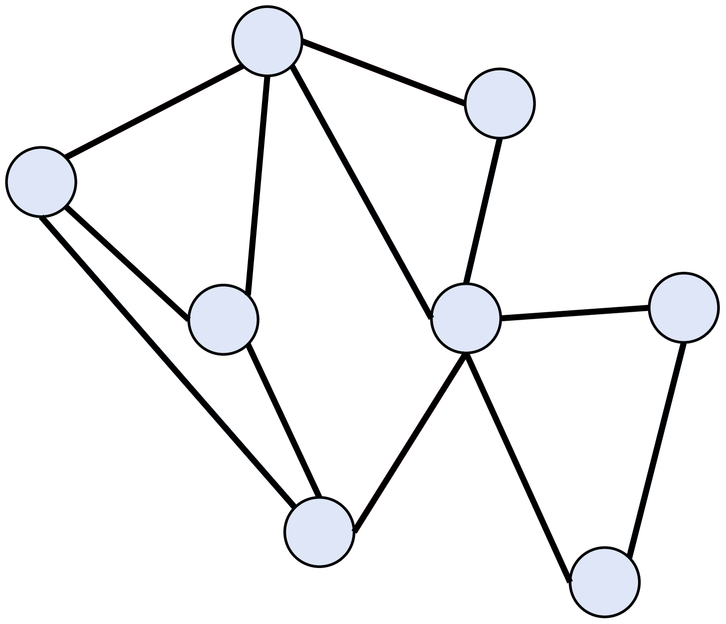
Linear Time $O(n + m)$ Solution

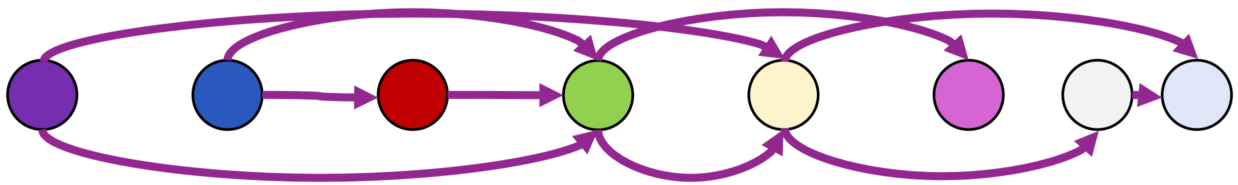
- Repeated peel vertex with minimum remaining induced degree and put in order



Linear Time $O(n+m)$ Solution

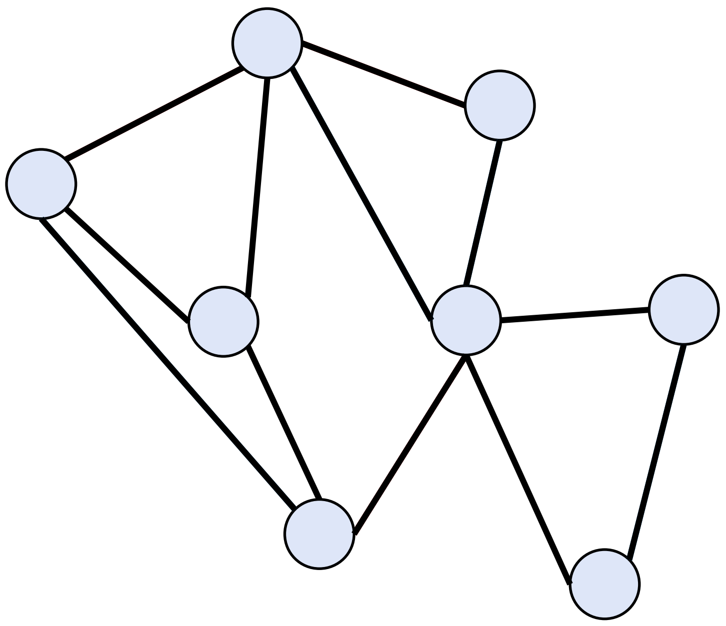
- Repeated peel vertex with minimum remaining induced degree and put in order



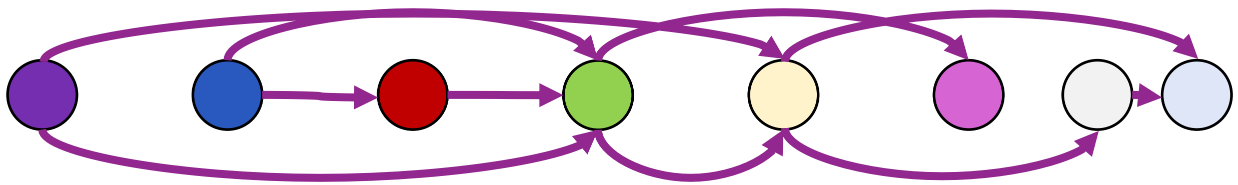


Linear Time $O(n + m)$ Solution

- Repeated peel vertex with minimum remaining induced degree and put in order



Outdegree: 2



Can make linear time by using n buckets in $O(n)$ space

Degeneracy of the Graph

- This property is called the **degeneracy** of the graph

Degeneracy of the Graph

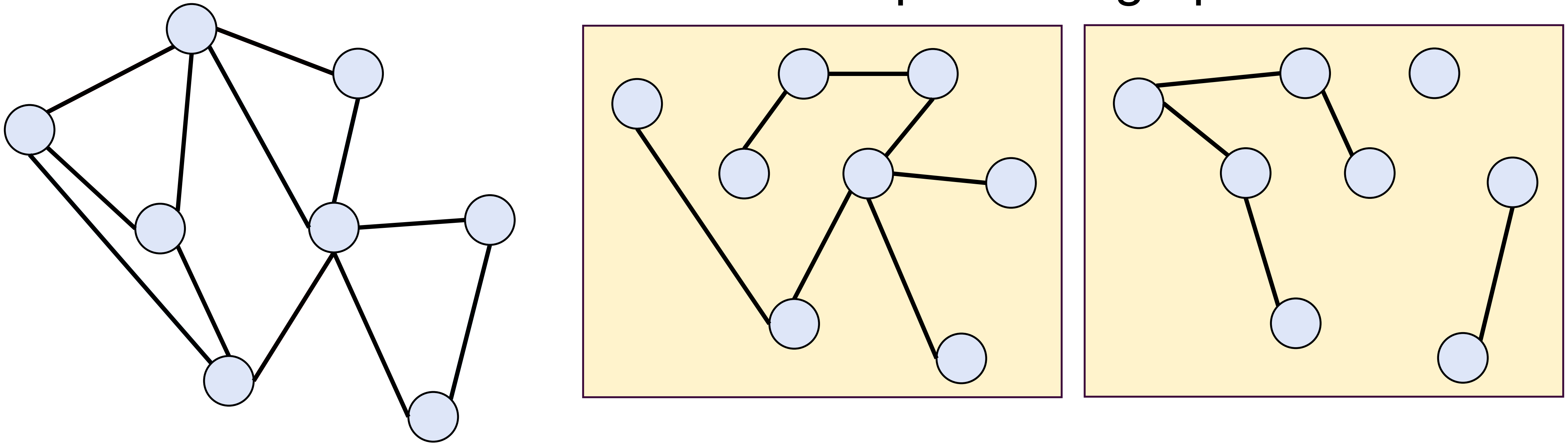
- This property is called the **degeneracy** of the graph
- Closely related to another concept called the **arboricity** of the graph

Degeneracy of the Graph

- This property is called the **degeneracy** of the graph
- Closely related to another concept called the **arboricity** of the graph
    - Minimum number of forests to decompose the graph

Degeneracy of the Graph

- This property is called the **degeneracy** of the graph
- Closely related to another concept called the **arboricity** of the graph
  - Minimum number of forests to decompose the graph



Degeneracy of the Graph

- This property is called the **degeneracy** $\boldsymbol{d}$ of the graph
- Closely related to another concept called the **arboricity** $\boldsymbol{\alpha}$ of the graph
  - Minimum number of forests to decompose the graph

$$\frac{\boldsymbol{d}}{\mathbf{2}} \leq \boldsymbol{\alpha} \leq \boldsymbol{d}$$

Degeneracy of the Graph

- This property is called the **degeneracy** $\boldsymbol{d}$ of the graph
- Closely related to another concept called the **arboricity** $\boldsymbol{\alpha}$ of the graph
  - Minimum number of forests to decompose the graph

$$\frac{\boldsymbol{d}}{\boldsymbol{2}} \leq \boldsymbol{\alpha} \leq \boldsymbol{d}$$

By Nash-Williams Theorem:

$$\boldsymbol{\alpha} = \max_{\boldsymbol{s}} \left\{ \left\lceil \frac{\boldsymbol{m_s}}{\boldsymbol{n_s} - \boldsymbol{1}} \right\rceil \right\}$$

Degeneracy is Very Small in Real-Life

Graph	Num. Vertices	Num. Edges	*d*
dblp	425,957	2,099,732	113
brain-network	784,262	267,844,669	1200
wikipedia	1,140,149	2,787,967	124
youtube	1,138,499	5,980,886	51
stackoverflow	2,601,977	28,183,518	198
livejournal	4,847,571	85,702,474	372
orkut	3,072,627	234,370,166	253
usa-central	14,081,816	16,933,413	3
usa-road	23,072,627	28,854,312	3
twitter	41,652,231	1,202,513,046	2488
friendster	65,608,366	1,806,067,135	304

Triangle Counting in Bounded Arboricity Graphs

- Combinatorial triangle counting conjecture: $\Omega(mn)$ time

Triangle Counting in Bounded Arboricity Graphs

- Combinatorial triangle counting conjecture: $\Omega(mn)$ time
- Can use matrix multiplication in n^ω time (compared to n^3 time)

Triangle Counting in Bounded Arboricity Graphs

- Combinatorial triangle counting conjecture: $\Omega(mn)$ time
- Can use matrix multiplication in n^{ω} time (compared to n^3 time)
- Bounded arboricity graphs: **$\boldsymbol{O(m\alpha)}$ time**

Triangle Counting in Bounded Arboricity Graphs

- Combinatorial triangle counting conjecture: $\Omega(mn)$ time
- Can use matrix multiplication in n^ω time (compared to n^3 time)
- Bounded arboricity graphs: $\boldsymbol{O(m\alpha)}$ **time**

Simple Algorithm (Chiba-Nishizeki '85)

1. For every edge, count number of triangles **incident to lower degree endpoint**
2. Divide total count by 3

Triangle Counting in Bounded Arboricity Graphs

- Proof of $\boldsymbol{O(m\alpha)}$ **time:**

<u>Simple Algorithm (Chiba-Nishizeki '85)</u>

1. For every edge, count number of triangles **incident to lower degree endpoint**
2. Divide total count by 3

- Equivalent to showing $\sum_{e=(u,v)\in E} \min\big(d(u), d(v)\big) \leq 2m\alpha$

Triangle Counting in Bounded Arboricity Graphs

- Proof of $O(m\alpha)$ **time:**

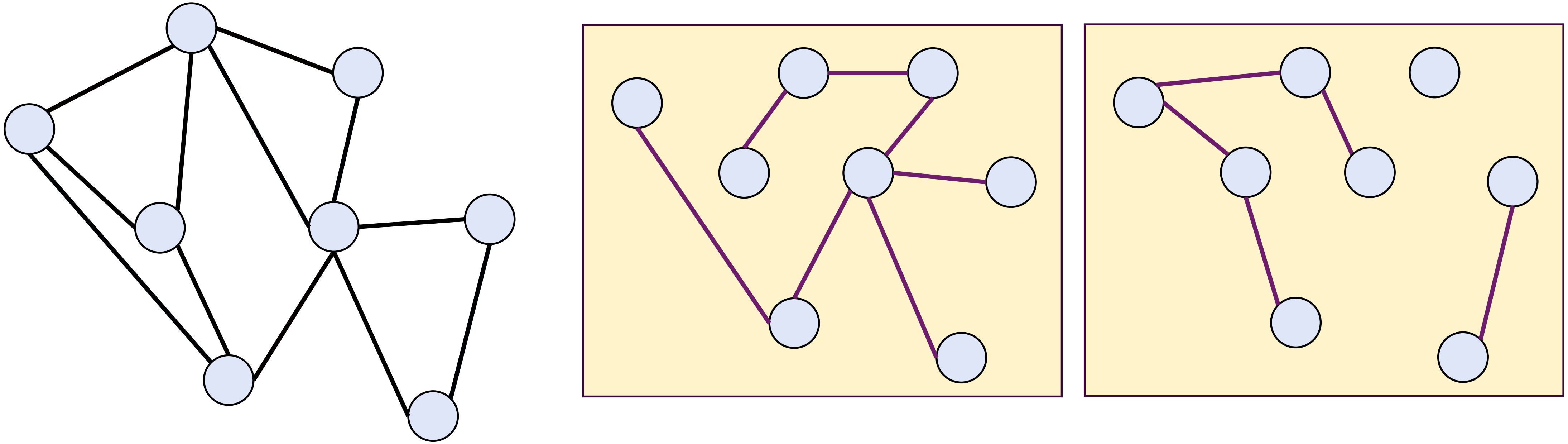
<u>Simple Algorithm (Chiba-Nishizeki '85)</u>

1. For every edge, count number of triangles **incident to lower degree endpoint**
2. Divide total count by 3

- Equivalent to showing $\sum_{e=(u,v)\in E} \min(d(u), d(v)) \leq 2m\alpha$
- Consider the arboricity decomposition of the graph

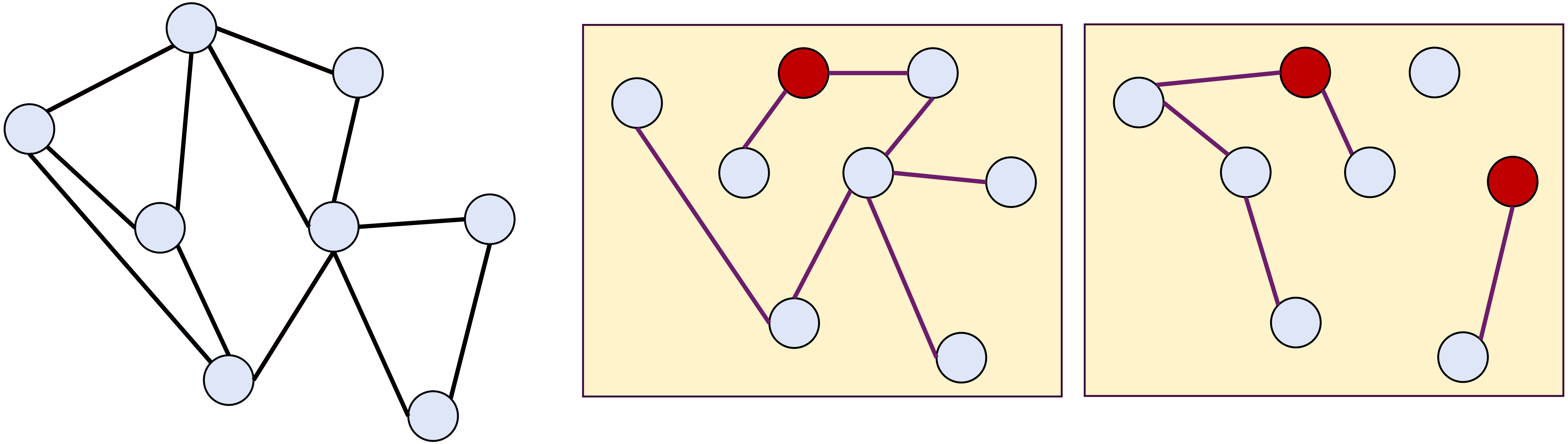
Triangle Counting in Bounded Arboricity Graphs

- Proof of $O(m\alpha)$ **time:**
  - Equivalent to showing $\sum_{e=(u,v)\in E} \min(d(u), d(v)) \leq 2m\alpha$
  - Consider the arboricity decomposition of the graph



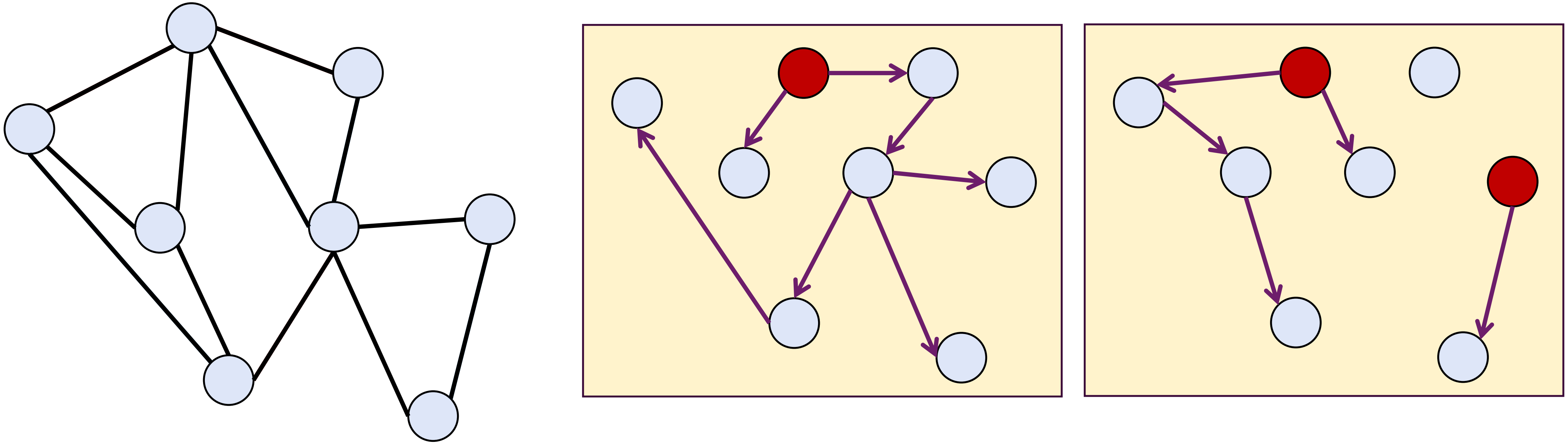
Triangle Counting in Bounded Arboricity Graphs

- Proof of $O(m\alpha)$ **time:**
  - Equivalent to showing $\sum_{e=(u,v)\in E} \min(d(u), d(v)) \leq 2m\alpha$
  - Consider the arboricity decomposition of the graph



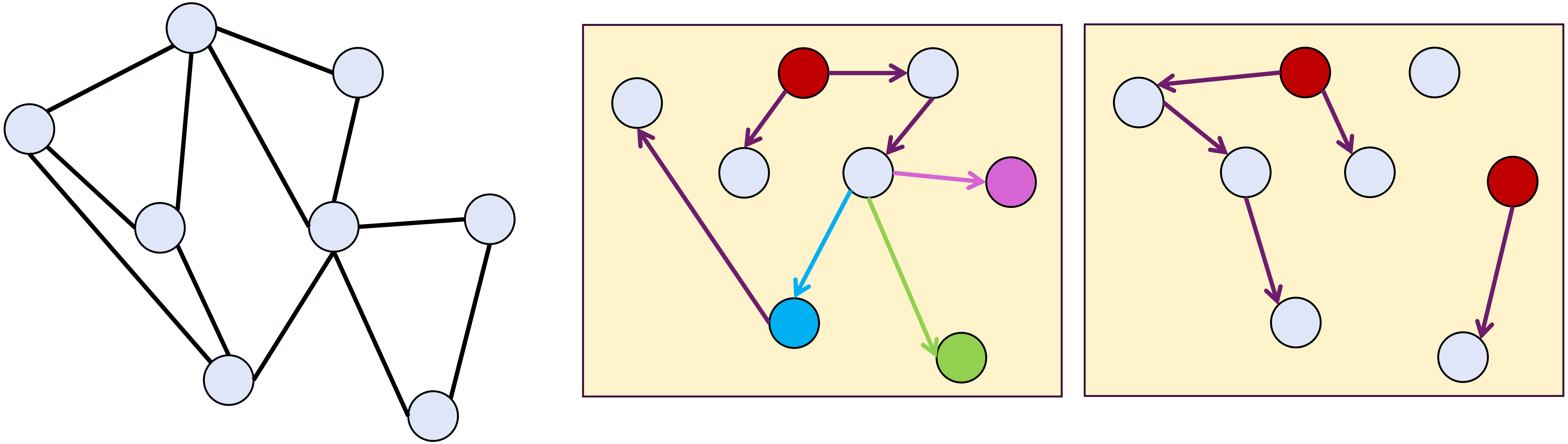
Triangle Counting in Bounded Arboricity Graphs

- Proof of $O(m\alpha)$ **time:**
  - Equivalent to showing $\sum_{e=(u,v)\in E} \min\big(d(u), d(v)\big) \leq 2m\alpha$
  - Consider the arboricity decomposition of the graph



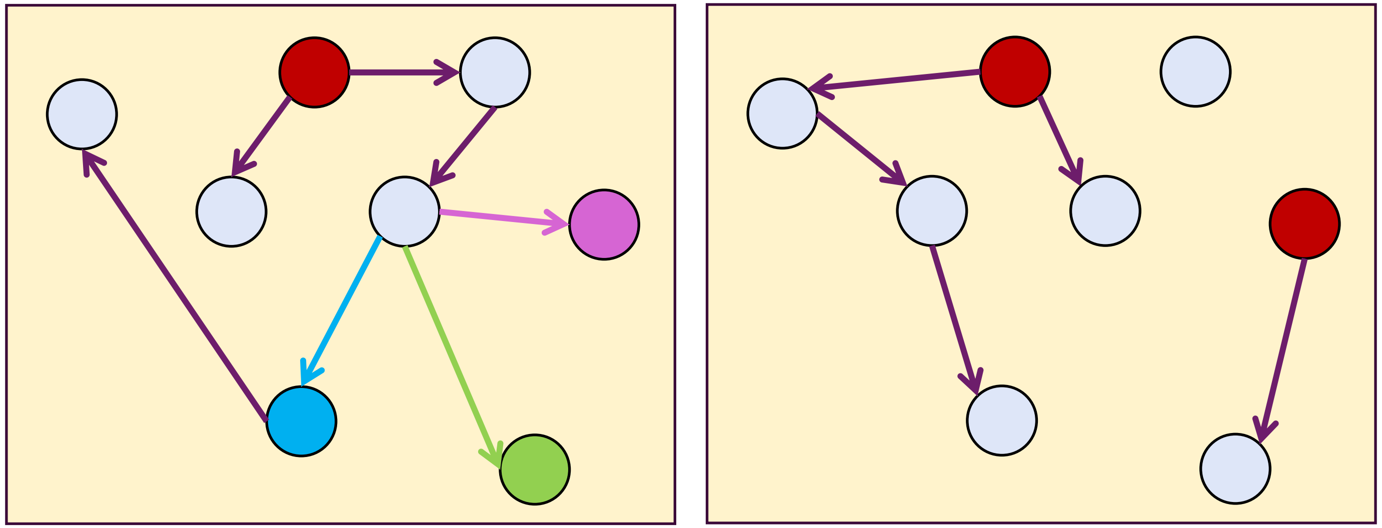
Triangle Counting in Bounded Arboricity Graphs

- Proof of $\boldsymbol{O(m\alpha)}$ **time:**
  - Equivalent to showing $\sum_{e=(u,v)\in E} \min\left(d(u), d(v)\right) \leq 2m\alpha$
  - Consider the arboricity decomposition of the graph



Triangle Counting in Bounded Arboricity Graphs

- Proof of **$O(m\alpha)$ time:**
  - Equivalent to showing $\sum_{e=(u,v)\in E} \min(d(u), d(v)) \leq 2m\alpha$
  - Consider the arboricity decomposition of the graph
  - Each vertex in F_i has **one edge**



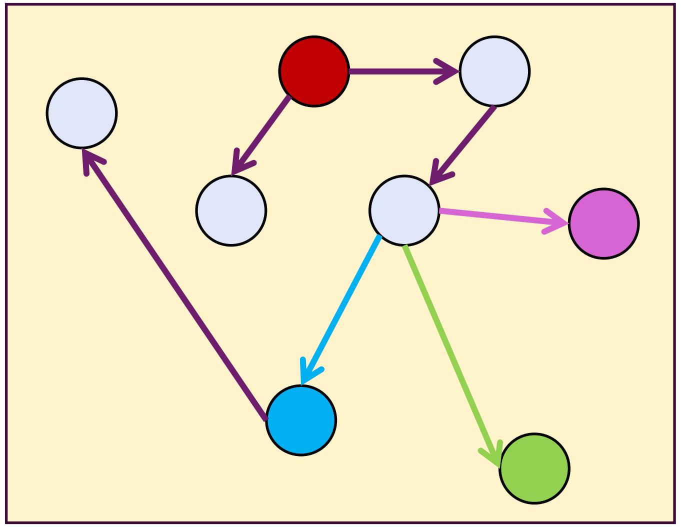
Triangle Counting in Bounded Arboricity Graphs

- Proof of $\boldsymbol{O(m\alpha)}$ **time:**
  - Equivalent to showing $\sum_{e=(u,v)\in E} \min\left(d(u), d(v)\right) \leq 2m\alpha$
  - Consider the arboricity decomposition of the graph
  - Each vertex in F_i has **one edge**

$$\sum_{(u,v)\in E} \min\left(\deg(u), \deg(v)\right)$$

$$\leq \sum_{1\leq i\leq \alpha} \sum_{e\in F_i} \deg\left(to(e)\right)$$

Follows because
$\deg\left(to(e)\right) \geq \min(\deg(u), \deg(v))$



Triangle Counting in Bounded Arboricity Graphs

- Proof of $\boldsymbol{O(m\alpha)}$ **time:**
  - Equivalent to showing $\sum_{e=(u,v)\in E} \min(d(u), d(v)) \leq 2m\alpha$
  - Consider the arboricity decomposition of the graph
  - Each vertex in F_i has **one edge**

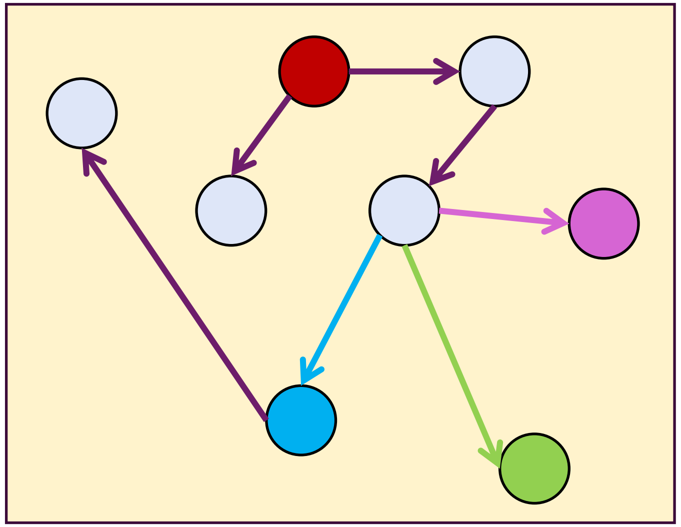
$$\sum_{(u,v)\in E} \min(\deg(u), \deg(v))$$

$$\leq \sum_{1\leq i\leq\alpha} \sum_{e\in F_i} \deg(to(e))$$

Follows because $\deg(to(e)) \geq \min(\deg(u), \deg(v))$

$$\leq \sum_{1\leq i\leq\alpha} \sum_{v\in V} \deg(v)$$

Since each vertex has at most one edge associated with it



Triangle Counting in Bounded Arboricity Graphs

- Proof of $\boldsymbol{O(m\alpha)}$ **time:**
  - Equivalent to showing $\sum_{e=(u,v)\in E} \min(d(u), d(v)) \leq 2m\alpha$
  - Consider the arboricity decomposition of the graph
  - Each vertex in F_i has **one edge**

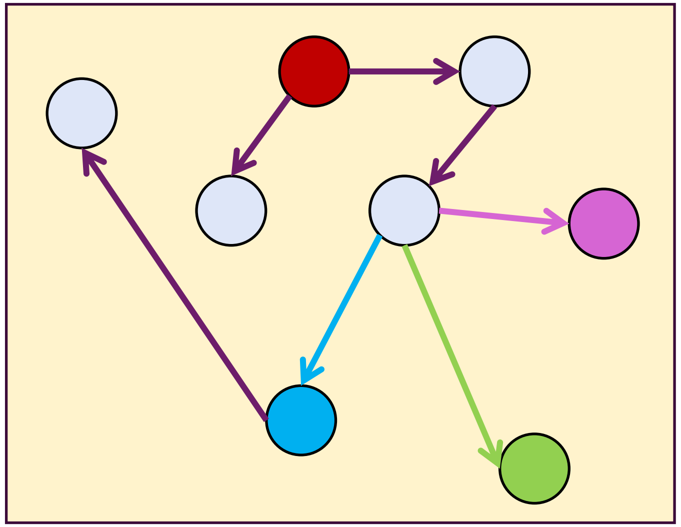
$$\sum_{(u,v)\in E} \min(\deg(u), \deg(v))$$

$$\leq \sum_{1\leq i\leq \alpha} \sum_{e\in F_i} \deg(to(e))$$

Follows because $\deg(to(e)) \geq \min(\deg(u), \deg(v))$

$$\leq \sum_{1\leq i\leq \alpha} \sum_{v\in V} \deg(v) = 2\alpha m$$

Since each vertex has at most one edge associated with it, the $\deg(to(e))$ of vertex $to(e)$ is counted at most once



Next Time

Bounded arboricity graphs in streaming model